MDPI

Article

A systematic evaluation and benchmarking of summarization methods for biomedical literature: From Classical Models to LLMs

Fabio Baumgärtel [1], Enrico Bono [1,2], Paul Perco [1,3] and Matthias Ley [1,2]*

[1] Delta4 GmbH, Vienna, Austria
[2] Division of Pediatric Nephrology and Gastroenterology, Department of Pediatrics and Adolescent Medicine, Comprehensive Center for Pediatrics, Medical University Vienna, Vienna, Austria
[3] Department of Internal Medicine IV, Medical University Innsbruck, Innsbruck, Austria
* Correspondence: matthias.ley@delta4.ai

Abstract

<TBD>

Draft: The exponential growth of scientific literature has created an urgent need for reliable tools that can distill research articles into concise and accurate summaries. In this study, we benchmarked 62 summarization models on 1,000 scientific abstracts with corresponding author-provided highlights as reference summaries. We evaluated models ranging from classical approaches – such as frequency-based and TextRank methods – to modern state-of-the-art language models, using a comprehensive set of surface-level, embedding-based and performance metrics. The results indicate that general-purpose models appear to outperform both reasoning-oriented and domain-specific models, whether tailored on biomedical text or designed for text summarization, with mid-sized general models outperforming larger proprietary ones. By systematically comparing traditional and modern approaches, this work highlights their respective strengths and limitations, offering insights into effective strategies for accelerating knowledge discovery in the molecular sciences.

A single paragraph of about 200 words maximum. For research articles, abstracts should give a pertinent overview of the work. We strongly encourage authors to use the following style of structured abstracts, but without headings: (1) Background: place the question addressed in a broad context and highlight the purpose of the study; (2) Methods: describe briefly the main methods or treatments applied; (3) Results: summarize the article's main findings; (4) Conclusions: indicate the main conclusions or interpretations. The abstract should be an objective representation of the article, it must not contain results which are not presented and substantiated in the main text and should not exaggerate the main conclusions.

Keywords: benchmarking; natural language processing; text summarization; largue language models; biomedical literature

Received:
Revised:
Accepted:
Published:

Citation: . Title. *Int. J. Mol. Sci.* **2025**, *1*, 0. https://doi.org/

1. Introduction

The exponential growth of scientific literature has created a demand for text summarization methods to support scientists in finding relevant information efficiently. Automatic text summarization (ATS) methods have evolved from statistical approaches to deep

learning-based models, becoming increasingly sophisticated and reliable at capturing essential parts from complex research articles. ATS methods have been previously evaluated and described [1,2], but few are tailored for scientific literature summarization [3,4].

The pre-neural era of text summarization was mainly characterized by extractive approaches, where in an unsupervised way, summaries were generated by using word or concept frequencies to identify relevant sentences. The first word-frequency based approaches were discussed by Luhn [5], who presented a method based on the assumption that recurrent words in a text are likely more important. Later, Edmunson [6] introduced concepts such as cue words, title words, and sentence position to further enhance the automatic summarization process. The concept of Term Frequency–Inverse Document Frequency (TF-IDF) was later adopted [7] and applied to text summarization by representing sentences as term-weight vectors that down-weight common terms and on the other hand up-weight rare terms that might be of more relevance. Thus, word-frequency based approaches have been extensively used in scientific text summarization, being at the basis of more sophisticated strategies [8]. Lastly, graph-based methods were implemented, where sentences were represented as nodes and relations between sentences, calculated by using similarity measures (i.e cosine similarity of TF-IDF vectors), as edges. Two graph-based methods gained popularity in the biomedical domain: TextRank, that builds a graph by breaking down the documents into single sentences to then apply the PageRank algorithm to assign importance scores to sentences. The summary is then generated by using the top-ranked sentences [9,10]. LexRank, instead, uses eigenvector centrality to find the most influenctial ones in the graph [11].

With the advent of Sequence-to-Sequence (Seq2seq) frameworks, summarization shifted toward neural approaches that paraphrase and condense text using Encoder-Decoder architectures, originally implemented with Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRUs) [12,13]. The introduction of self-attention mechanisms replaced RNNs by processing sequences in parallel rather than sequentially, enabling the capture of complex linguistic patterns and long-range contextual relationships [14]. This innovation laid the foundation for transformer architectures that quickly gained popularity in performing a wide range of Natural Language Processing (NLP) tasks, including text summarization. One of the earliest and most influential transformer-based models, Bidirectional Encoder Representations from Transformers (BERT) [15], was widely adopted in domain specific tasks thanks to the possibility of fine-tuning by adding a task-specific output layer. Inspired by BERT's architecture, several abstractive summarization models emerged, including Bidirectional and Auto-Regressive Transformer (BART) - a denoising autoencoder for pretraining sequence-to-sequence models [16] that can be trained or fine-tuned on scientific literature [17,18]. The Text-to-Text Transfer Transformer (T5) model was introduced as a unified text-to-text framework for a broad spectrum of NLP tasks due to its high flexibility with no need for architectural changes [19]. Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence (PEGASUS), was specifically proposed for abstractive summarization [20] and has been adapted for scientific text with domain-specific variants including "google/pegasus-pubmed" and "google/bigbird-pegasus-large-pubmed". Robustly Optimized BERT Approach (RoBERTa) is an optimized version of BERT trained on a bigger corpus of text, which led to the creation of Longformer [21], a transformer-based architecture that can handle longer texts for text-to-text generation, with "allenai/led-base-16384" and "led-large-16384-arxiv" as notable examples [22]. Despite these advances, the field of ATS quickly moved towards decoder-only architectures which are at the basis of LLMs, able to capture semantic relations with higher flexibility and specificity. LLMs can be classified as (i) general-purpose models, which leverage their broad domain knowledge

across diverse NLP tasks, (ii) reasoning-oriented models, characterized by logical text understanding through iterative chain-of-thought processing and instruction tuning [23], and (iii) domain-specific models, tailored for specialized tasks or scientific domains. Several families of LLMs have been developed, including the GPT series developed by OpenAI (GPT-1 [24] through GPT-5) and open-source variants like GPT:OSS, all pre-trained on large-scale text corpora through self-supervised learning. Similarly, Anthropic's Claude Models are built on transformer architecture and trained through a Constitutional AI approach [25]. This family also includes a series of reasoning models such as Sonnet-4 and Opus-4. Meta's Llama family, with LLaMA 3.1 as the most capable open-source model available to date, includes domain-specific adaptations such as OpenBioLLM-LLaMA-3, a biomedical variant trained on a large corpus of high-quality biomedical data, and MedLLaMA-2, a medical language model based on LLaMA 2 architecture. Google developed a series of lightweight models including the Gemma series [26], with Gemma3 as its latest and most powerful reasoning model. Microsoft introduced the Phi series, which comprises Phi-4-reasoning and Phi-4-mini-reasoning, alongside BioGPT [27], a domain-specific model built on the GPT architecture and fine-tuned for biomedical applications. IBM released the Granite series, with Granite 4.0 as its reasoning-capable variant. Mistral AI developed the Mistral family, including Magistral as its first reasoning model, and Biomistral, an open-source variant pretrained on PubMed Central data for biomedical text processing. Alibaba Cloud introduced the Qwen 3 series as an open-source LLM family, which inspired SciLitLLM, a specialized model for scientific literature understanding based on Qwen2.5 and trained through continual pre-training (CPT) and supervised fine-tuning (SFT) on scientific literature [28]. DeepSeek has developed reinforcement learning (RL)-driven reasoning models that achieve performance comparable to state-of-the-art closed-source models while requiring only a fraction of their training costs [29]. Lastly, APERTUS represents Switzerland's first large-scale open, multilingual language model with a fully documented and openly accessible development process.

To the best of our knowledge, no comprehensive benchmarking of text summarizations models on biomedical literature has been performed to date. This study addresses this gap by systematically evaluating 62 summarization models, ranging from classical approaches to state-of-the-art LLMs, on a curated dataset of 1,000 biomedical abstracts with corresponding highlight sections as reference studies. By identifying the strengths and limitations of each approach, we provide actionable insights for selecting appropriate summarization tools to accelerate knowledge discovery in biomedical sciences.

2. Materials and Methods

2.1. Gold-Standard Dataset

We generated a gold-standard benchmarking dataset comprising 1,000 biomedical peer-reviewed articles from *ScienceDirect* and *Cell Press* - as these publishers provide a standardized *Highlights* section for each publication [30,31]. This section provides concise bullet points capturing the main findings of each article. The concatenated highlights served as reference summaries in our evaluation, while the corresponding abstracts were used as input texts for the summarization task.

Articles were collected systematically across a variety of journals to ensure coverage of different fields within molecular sciences, including drug discovery, genomics, proteomics, biotechnology, and biochemistry. We selected 50 articles from each of 20 different journals across the two publishers, resulting in 1,000 papers as shown in Table 1.

Table 1. Overview of journals in the gold-standard dataset.

Publisher	Journal
ScienceDirect	Drug Discovery Today
ScienceDirect	Journal of Molecular Biology
ScienceDirect	FEBS Letters
ScienceDirect	Journal of Biotechnology
ScienceDirect	Gene
ScienceDirect	Genomics
ScienceDirect	Journal of Proteomics
ScienceDirect	The International Journal of Biochemistry & Cell Biology
ScienceDirect	Cytokine
ScienceDirect	Developmental Cell
Cell	Cell
Cell	Cancer Cell
Cell	Cell Chemical Biology
Cell	Cell Genomics
Cell	Cell Host & Microbe
Cell	Cell Metabolism
Cell	Cell Reports
Cell	Cell Reports Medicine
Cell	Cell Stem Cell
Cell	Cell Systems

This setup provides standardized pairs of abstracts and reference summaries that can be directly used for evaluating automatic summarization methods.

2.2. Summarization Methods

We evaluated 62 summarization models, ranging from simple frequency-based algorithms to state-of-the-art small- and large language models (SLMs & LLMs) as listed in Table 2.

The summarization models were grouped into five categories:

1. Traditional models: We included two traditional extractive models as baselines for comparison with newer, more complex approaches: a simple frequency-based method and TextRank [9].
2. Encoder-Decoder models (EDMs): We included a set of pre-trained encoder-decoder models available through the HuggingFace library, encompassing both general-purpose and domain-specific variants. The general-purpose group includes BART (base and large) [16], T5 (base and large) [32], mT5 [33], and a variety of PEGASUS models [20]. The domain-specific group includes PEGASUS and BigBird models fine-tuned on PubMed data as well as LED [21] (arXiv-tuned). These models are commonly applied for abstractive summarization tasks and represent well-established neural architectures within our benchmark.
3. General-purpose models: We also evaluated a range of widely used LLMs and SLMs, with SLMs being defined as <10 B parameters [34]. This group includes Gemma [26], Granite [35], LLaMA [36], Mistral [37], Phi [38,39], GPT [40,41], Claude [42], and Apertus [43], which represent the current landscape of general-purpose systems.
4. Reasoning-oriented models: We categorized both LLMs and SLMs with advanced reasoning capabilities as reasoning-oriented models. This group includes models from the DeepSeek-R1 family [44], Qwen [45], GPT variants such as GPT-oss [46] and GPT-5 [47], Magistral [48], and additional Claude models. Their design emphasizes multi-step problem solving, allowing us to explore whether reasoning capabilities affects summarization performance.

5. Domain-specific models: To assess whether domain adaptation improves summarization quality, we included both LLMs and SLMs fine-tuned on scientific/biomedical data. These include BioGPT [49], MedLLaMA2 [50], OpenBioLLM [51], BioMistral [52], and SciLitLLM1.5 models [53].

Table 2. Overview of summarization methods/models evaluated in this study, organized by category.

Category	Methods/Models
Traditional models	textrank; frequency
General-purpose EDMs	facebook/bart-base; google-t5/t5-base; google-t5/t5-large; google/pegasus-large
Domain-specific EDMs	facebook/bart-large-cnn; google/pegasusxsum; google/pegasus-cnn_dailymail; google/pegasus-pubmed; google/bigbird-pegasuslarge-pubmed; csebuetnlp/mT5_multilingual_XLSum; led_large_16384_arxiv_summarization
General-purpose SLMs	gemma3:270M; gemma3:1b; gemma3:4b; PetrosStav/gemma3-tools:4b; granite3.3:2b; granite3.3:8b; granite4:tiny-h; granite4:small-h; granite4:micro; granite4:micro-h; llama3.1:8b; llama3.2:1b; llama3.2:3b; mistral:7b; phi3:3.8b; gpt-4o-mini; gpt-4.1-mini; chat_swiss-ai/Apertus-8B-Instruct-2509
General-purpose LLMs	gemma3:12b; mistral-nemo:12b; mistral-small3.2:24b; mistral-small-2506; mistral-medium-2505; mistral-large-2411; phi4:14b; gpt-3.5-turbo; gpt-4o; gpt-4.1; claude-3-5-haiku-20241022
Reasoning-oriented SLMs	deepseek-r1:1.5b; deepseek-r1:7b; deepseek-r1:8b; qwen3:4b; qwen3:8b;
Reasoning-oriented LLMs	deepseek-r1:14b; gpt-oss:20b; gpt-5-nano-2025-08-07; gpt-5-mini-2025-08-07; gpt-5-2025-08-07; claude-sonnet-4-20250514; claude-opus-4-20250514; claude-opus-4-1-20250805; magistral-medium-2509
Domain-specific SLMs	completion_microsoft/biogpt; medllama2:7b; chat_aaditya/OpenBioLLM-Llama3-8B; conversational_BioMistral/BioMistral-7B; chat_Uni-SMART/SciLitLLM1.5-7B
Domain-specific LLMs	chat_Uni-SMART/SciLitLLM1.5-14B

With this selection, we covered models of different sizes and release dates, ensuring that both widely adopted and recent architectures were represented. Extraordinarily large models, such as LLaMA 3.1 405B, were excluded because their computational requirements exceed what is practical for typical summarization pipelines.

These 62 diverse models were all tasked with generating summaries for each of the 1,000 abstracts in the dataset, resulting in 62,000 generated summaries available for evaluation.

2.3. Prompt Design

To ensure comparability across models, we prompted all summarization systems with an identical task description. The prompt instructed the models to generate concise summaries focused on the main findings of each publication while excluding unnecessary background or methodological details. Each model received the publication title and abstract as input and was asked to produce an output of approximately 15–100 words. If

the abstract did not contain any substantive results or conclusions, the model was instructed to return the predefined token INSUFFICIENT_FINDINGS.

The exact prompt used for all models was as follows:

```
Summarize the provided publication (title and abstract) in 15-100 words.

Key requirements:
- Identify main findings, results, or contributions
- Preserve essential context and nuance
- Exclude background, methods unless crucial to conclusions
- Write concisely and objectively
- Avoid repetition and unnecessary qualifiers

If no substantial findings exist, respond: 'INSUFFICIENT_FINDINGS'
```

2.4. Evaluation Metrics

As there is no single metric that can fully reflect summary quality, especially in the biomedical field where both coverage of key information and factual correctness are critical, we employed multiple metrics grouped into two categories: traditional surface-level metrics and embedding-based metrics. By combining these metrics into one final overall score, we obtained a balanced benchmark that reflects summary quality.

2.4.1. Surface-level Metrics

Surface-level metrics compare the generated summaries with the reference summaries mainly at the word or phrase level. While they do not capture meaning beyond surface overlap, they remain common metrics in summarization research and provide a straightforward foundation for evaluation. We used three ROUGE variants (ROUGE-1, ROUGE-2, ROUGE-L) [54], BLEU [55], and METEOR [56]. ROUGE-1 and ROUGE-2 measure how many unigrams (single words) or bigrams (word pairs) from the reference appear in the generated output, while ROUGE-L identifies the longest sequence of words shared between the two. BLEU calculates how many n-grams in the output also occur in the reference, emphasizing precision over recall and applying a brevity penalty to counteract the tendency toward overly short summaries. METEOR extends n-gram matching by considering word stems and synonyms, making it more robust to wording variations. Together, these metrics offer a simple but transparent point of reference.

2.4.2. Embedding-based Metrics

To capture similarity beyond surface-level word overlap, we included a set of embedding-based metrics built on pre-trained transformer models. These methods generate vector representations of text, allowing them to capture semantic similarity rather than just word overlap. We employed RoBERTa [57] and DeBERTa [58], two transformer-based models with strong performance across NLP tasks. In summarization evaluation, they can assess whether two summaries capture the same content even if phrased differently.

We further included all-mpnet-base-v2 [59], a transformer model fine-tuned for sentence similarity. Unlike RoBERTa and DeBERTa, which are general-purpose encoders, MPNet was trained with a focus on alignment at the sentence-level. This characteristic makes it a useful complement to the other metrics, as it is particularly sensitive to whether the overall meaning of a reference summary is preserved in the system output.

Finally, to evaluate factual consistency, we applied AlignScore [60], a metric designed to assess whether the statements in a generated summary are supported by the source text. In contrast to the other metrics, AlignScore compares the output to the input text itself (i.e.

the publication abstract) rather than the reference summary (i.e. the Highlights section), as factual accuracy can only be assessed relative to the original input text. This addition ensures that our evaluation captures errors and hallucinations that might otherwise be overlooked.

2.4.3. Ranks Calculation

For each of the evaluation metric, ranks were calculated by assigning 1 to the lowest metric value, reflecting the best-performing model. Different weights were then assigned to each score (Table 3) to compute a weighted overall score by multiplying each metric rank by its weight and summing all weighted ranks. Thus, the overall performance rank was constructed by prioritizing the semantic metrics (RoBERTa and DeBERTa), which better reflect true summary quality, over the lexical (ROUGE, BLEU, METEOR) and alignment-based (all-mpnet-base-v2, AlignScore) metrics.

Table 3. Overview of the assigned weights for each evaluation metric.

Evaluation Metric	Weight	Percentage
RoBERTa	0.25	50%
DeBERTa	0.25	
ROUGE-1	0.07	30%
ROUGE-2	0.07	
ROUGE-L	0.05	
METEOR	0.06	
BLEU	0.05	
all-mpnet-base-v2	0.11	20%
AlignScore	0.09	

2.4.4. Performance Metrics

In addition to summary quality, we also considered some practical aspects of model performance:

- Execution time records the average time required to generate summaries, which is critical when processing large datasets.
- Length compliance (% within bounds) measures how often the generated summaries fall within the target length range specified in the prompt. It reflects a model's ability to follow explicit output-length instructions as it penalizes both overly short and excessively long responses.
- Insufficient findings describes how often a model returned the predefined token 'INSUFFICIENT_FINDINGS' instead of producing a summary, capturing cases where it concluded the input lacked substantive findings. Summaries that were clearly nonsensical, contained syntax errors, or no meaningful content were further flagged as 'INSUFFICIENT_FINDINGS' during post-processing.

These measures complement the quality metrics by addressing whether a method is not only accurate but also feasible for real-world use.

2.5. Benchmarking Framework

The benchmark was conducted using Python 3.12. Gold standard data were retrieved from open-access publications published by ScienceDirect and Cell Press through manual extraction of titles, abstracts, and highlight sections, along with metadata including publication URLs, identifiers, section types, and article types where available. All data were stored in machine-readable JSON format.

The framework was implemented using the Python standard library supplemented by several specialized packages: pandas [61] for data import and export, scikit-learn [62] for computing cosine similarities of embeddings and TF-IDF vectors, networkx [63] for graph construction and PageRank algorithm [64]. Additional evaluation metrics were computed using NLTK [65] for METEOR and BLEU scores, ROUGE-score, BERT-score [66], AlignScore, and sentence-transformers [67] with the all-mpnet-base-v2 model.

Communication with proprietary closed-source LLMs was facilitated through the official Python APIs provided by Anthropic, Mistral AI, and OpenAI. Local LLM execution was performed on a workstation equipped with a NVIDIA RTX A4000 GPU (16GB VRAM) running Ollama as a backend service, accessed through its Python API along with the transformers library [68].

All LLMs were configured with a temperature parameter of 0.2 to optimize reproducibility while avoiding completely deterministic outputs. For the latest generation of OpenAI models featuring adaptive reasoning capabilities, the configuration was set to `text.verbosity = low` and `reasoning.effort = minimal`. The full set of parameters and prompts are documented in the `config.py` file in the GitHub repository.

2.6. Data Availability

The complete source code, documentation, gold standard dataset, and processed results are available at:

https://www.github.com/Delta4AI/LLMTextSummarizationBenchmark.

3. Results

Our benchmark results offer a comparative view of summarization performance across all evaluated models. We first present overall rankings, followed by comparisons between different model groups. Additionally, we examine results on individual metrics, runtime performance, and correlations between the evaluation metrics used.

3.1. Overall Model Performance

Based on the performance outcomes shown in Figure 1, models from the Mistral family occupied the top positions of the ranking, achieving strong performance across the majority of surface-level metrics (ROUGE-1, ROUGE-2, ROUGE-L, METEOR, BLEU) and embedding-based measures (RoBERTa, DeBERTa, all-mpnet-base-v2, AlignScore). According to the performance rank, mistral-medium-2505 ranks first, followed by mistral-small-2506, mistral-small-3.2:24b, and mistral-large-2411.

The lowest-ranked models include pegasus-xsum, pegasus-pubmed and bigbird-pegasus-large-pubmed from the Pegasus family, with the latter being the worst-performing model. Domain-specific models such as OpenBioLLM-Llama3-8B, biogpt, and multilingual_XLSum, show poor performance across all the metrics.

Among the 10 top ranked models, five are general-purpose LLMs, three are general-purpose SLMs, and two are reasoning-oriented LLMs. A similar trend is present by looking at the top half of the ranking (positions 10 to 32), except from the presence of a single domain-specific LLM ranked at 20 (SciLitLLM1.5-14B). In contrast, in the lower half of the ranking, where models start to perform poorly across most metrics, the majority are reasoning-oriented SLMs, general-purpose EDMs, domain-specific EDMs, and traditional models. The best and worst models by category are reported in Table 4, while those by model family are reported in Table 5.

Table 4. Overview of the best- and worst-performing models by category. Only categories with at least 3 models are reported.

Category	Best Model	Rank	Worst Model	Rank
General-purpose EDMs	T5-large	48	Pegasus-large	55
Domain-specific EDMs	led_large_16384_arxiv_summarization	46	bigbird-pegasus-large-pubmed	62
General-purpose SLMs	GPT-4o-mini	6	Phi3:3.8b	45
General-purpose LLMs	Mistral-medium-2505	1	Mistral-nemo:12b	30
Reasoning-oriented SLMs	qwen3:8b	35	Deepseek-r1:8b	50
Reasoning-oriented LLMs	GPT-5-nano	7	GPT-5	41
Domain-specific SLMs	SciLitLLM1.5-7B	37	OpenBioLLM-Llama3-8B	61

Table 5. Overview of the best- and worst-performing models by family. Only families with at least 3 models are reported.

Family	Best Model	Rank	Worst Model	Rank
Pegasus	pegasus-cnn_dailymail	49	bigbird-pegasus-large-pubmed	62
T5	T5-large	48	mT5_multilingual_XLSum	60
Qwen	SciLitLLM1.5-14B	20	Qwen3:4b	43
Gemma	Gemma3-tools:4b	10	Gemma3:270M	42
Granite	Granite3.3:8b	9	Granite3.3:2b	28
LLaMA	Llama3.2:3b	26	OpenBioLLM-Llama3-8B	61
Mistral	Mistral-medium-2505	1	BioMistral-7B	38
Phi	Phi4:14b	23	Phi3:3.8b	45
DeepSeek	Deepseek-r1:14b	34	Deepseek-r1:8b	50
GPT	GPT-4o	5	BioGPT	59
Claude	Claude-sonnet-4	8	Claude-opus-4-1	33

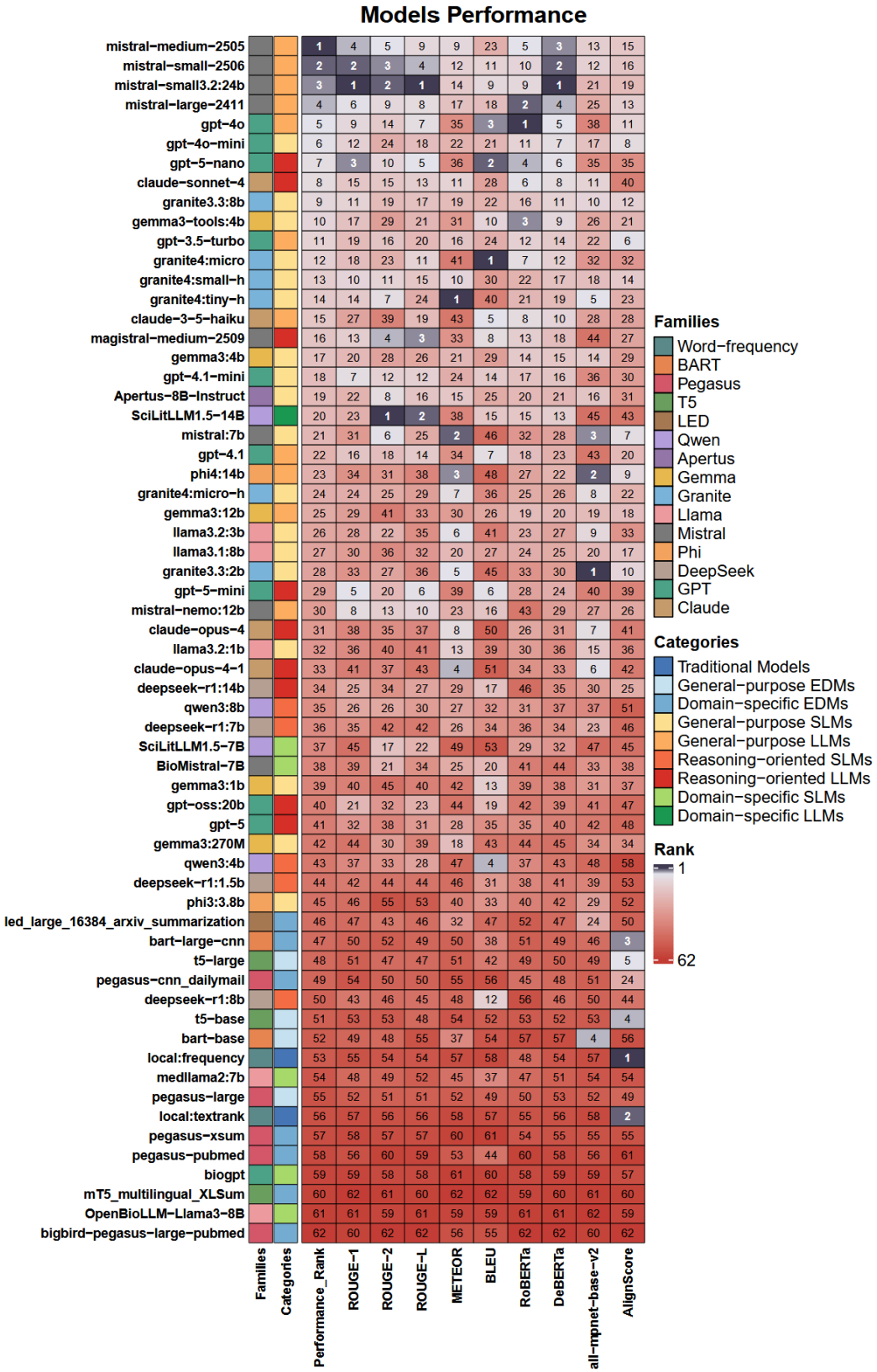


Figure 1. Overview of the performance of all evaluated models across all surface-level and embedding-based metrics. Each row corresponds to one model, and each column to a specific metric, with lower ranks indicating better performance. The figure displays each model's family (e.g., GPT, DeepSeek, Gemma, Granite), and category (e.g., encoder–decoder, general-purpose SLMs, reasoning-oriented LLMs). Models are sorted by their weighted average rank across metrics (Performance_Rank), where lower ranks indicate better performance.

3.2. Group Comparisons

Figure 2 summarizes the average performance of the nine model categories based on the overall metric mean score. General-purpose LLMs achieved the highest mean score (0.527), followed by general-purpose SLMs (0.519) and reasoning-oriented LLMs (0.515). Traditional extractive models, general-purpose EDMs, and domain-specific EDMs performed considerably lower, with mean scores of 0.451, 0.471, and 0.410, respectively. The domain-specific SLMs showed weak overall performance (0.439), whereas the domain-specific LLMs achieved a higher score (0.513; single model).

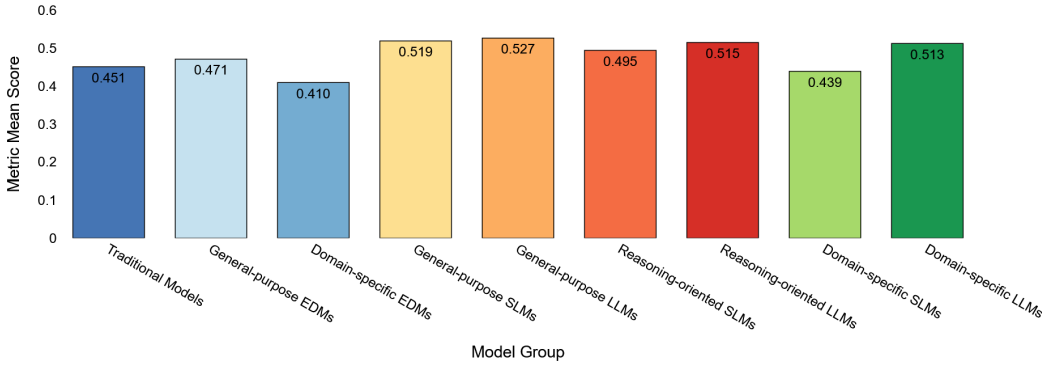
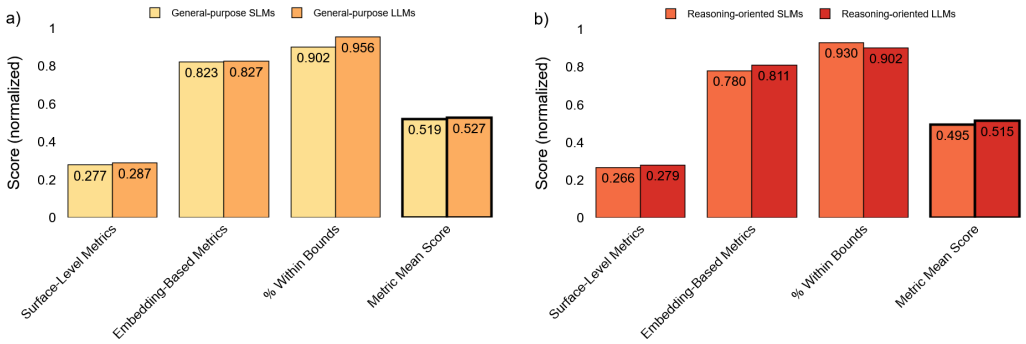


Figure 2. Average metric mean score across the nine model categories. The figure highlights clear performance differences between categories, with general-purpose LLMs performing best overall, followed by general-purpose SLMs and reasoning-oriented LLMs. Traditional models, EDMs, and domain-specific SLMs achieved notably lower scores.

3.2.1. SLMs vs. LLMs

To further analyze differences between small and large language models, we compared the performance of SLMs and LLMs within both the general-purpose and reasoning-oriented groups (Figure 3). In both categories, LLMs achieved higher overall metric mean scores (0.527 vs 0.519 and 0.515 vs. 0.495) and generally performed better on surface-level and embedding-based metrics. Compliance with word-length bounds favored LLMs in the general-purpose group but SLMs in the reasoning-oriented group. The comparison for domain-specific models is omitted, as this category includes only a single LLM, preventing meaningful comparison.

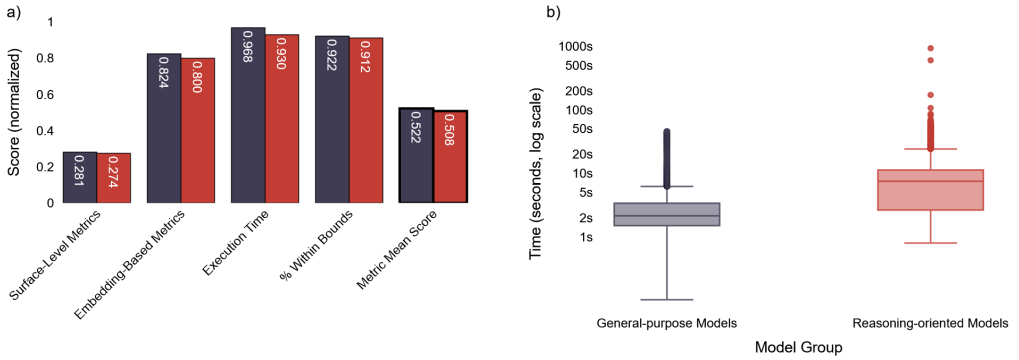


(**a**) General-purpose SLMs vs. LLMs. (**b**) Reasoning-oriented SLMs vs. LLMs.

Figure 3. (**a**) Comparison between general-purpose SLMs and LLMs across key evaluation metrics. (**b**) Comparison between reasoning-oriented SLMs and LLMs. In both groups, LLMs achieved slightly higher overall metric mean scores, while SLMs occasionally performed better on individual metrics.

3.2.2. General-purpose Models vs. Reasoning-oriented Models

Figure 4a compares the two largest and most competitive groups (general-purpose and reasoning-oriented models) across multiple evaluation aspects, including both SLMs and LLMs. General-purpose models achieved slightly higher scores across surface-level metrics, embedding-based metrics, execution time, compliance with word-length bounds, and overall metric mean score. The largest difference was observed in execution time, where general-purpose models reached a score of 0.968 compared to 0.930 for reasoning-oriented models. Figure 4b provides a detailed view of these runtime differences. Smaller but consistent advantages were also seen in surface-level metrics, embedding-based metrics, and compliance with word-length bounds.



(**a**) General-purpose vs. reasoning-oriented models across key evaluation aspects.

(**b**) Execution time distribution for the same two groups.

Figure 4. (**a**) Comparison between general-purpose and reasoning-oriented models across key evaluation metrics. General-purpose models achieved higher scores across all categories, including surface-level and embedding-based metrics, execution time, compliance with word-length bounds, and overall Metric Mean Score. (**b**) Distribution of execution times for the same groups, showing that general-purpose models produced summaries more efficiently and with lower variability.

3.3. Metric Correlations

To examine how the different evaluation metrics relate to each other, we computed pairwise Pearson correlation coefficients across all models (Figure 5).

Strong positive correlations were observed among the surface-level metrics (ROUGE-1, ROUGE-2, ROUGE-L, METEOR, and BLEU). ROUGE variants showed almost identical behavior ($\rho > 0.9$), while BLEU and METEOR demonstrated slightly weaker but still substantial alignment with ROUGE measures.

Most embedding-based metrics (RoBERTa, DeBERTa, and all-mpnet-base-v2) showed very high internal consistency ($\rho > 0.8$), reflecting their shared focus on semantic similarity beyond surface-level overlap. When compared with surface-level metrics, correlations were moderate to strong ($\rho \approx 0.7$–1.0), indicating that both categories capture related but not identical dimensions of summary quality.

AlignScore correlated moderately with the other metrics ($\rho \approx 0.4$–0.7), which can be attributed to its different point of reference, as it compares generated summaries directly with source abstracts instead of the reference summaries like other metrics.

Overall, these relationships demonstrate that the various metrics are broadly consistent while providing complementary perspectives. This supports the use of an aggregated "Metrics Mean Score" as a balanced indicator of overall summarization performance.

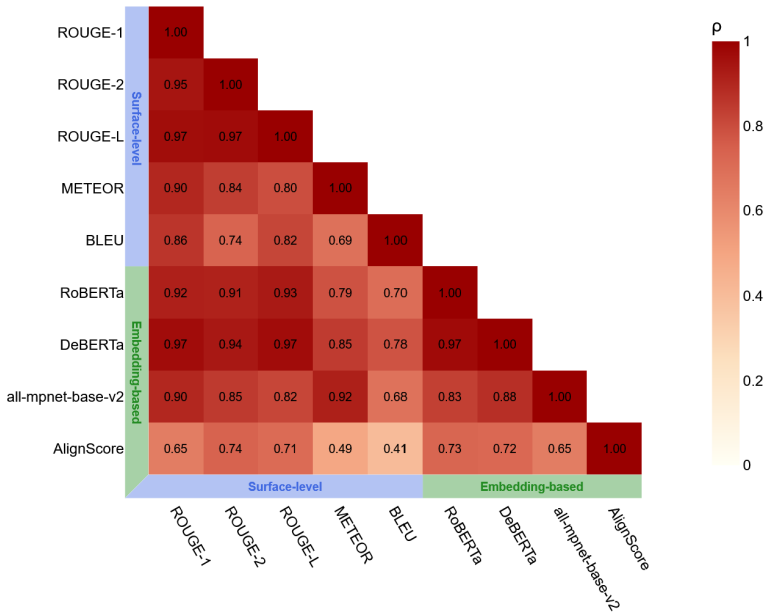


Figure 5. Correlation matrix of evaluation metrics. Each cell in the matrix represents the Pearson correlation coefficient (ρ) between two metrics based on their mean scores across all models. Surface-level and embedding-based metrics show strong internal consistency, while AlignScore exhibits lower correlations due to its distinct focus on factual consistency with source abstracts.

3.4. Qualitative Analysis: Case Study

To illustrate how concept coverage differs between models beyond aggregate metrics, we examine two summaries of a biomedical research article [doi:10.1016/j.cell.2025.05.026] against publisher-provided highlights. The source article, titled *"A percolation phase transition controls complement protein coating of surfaces"*, includes four key highlights: (H1) The complement protein network has a switch-like response when attacking surfaces; (H2) Complement "decides" to coat surfaces if surface protein spacing is below a threshold; (H3) Complement's threshold decision-making arises from a percolation phase transition; (H4) Complexity science shows how complement makes discrete decisions attacking surfaces.

The generated summaries are as follows:

- **mistral_medium-2505:** "This study reveals that the complement system, a network of blood proteins, exhibits a sharp activation threshold based on the surface density of attachment points on materials. This threshold behavior, observed across scales from nanoparticles to macroscale pathologies, arises from a percolation-type phase transition in a minimal subnetwork of complement proteins. The findings suggest that this critical transition acts as a switch, determining whether the complement system will interact with and initiate inflammation against a material."
- **SciLitLLM1.5-14B:** "A sharp threshold for complement activation upon varying a fundamental material parameter, the surface density of potential complement attachment points, was experimentally demonstrated and computationally explained."

Table 6 summarizes the concept-level coverage and overall performance for both models. While both models capture the surface density threshold (H2), the lower-performing model omits the mechanistic explanation (H3) and only partially conveys the behavioral characteristics (H1, H4), resulting in substantially lower concept coverage and semantic alignment. The overall performance scores, derived using the weighted metric aggregation introduced in Section 2.4.3, further reflect this difference, with **mistral_medium-2505** achieving 0.619 compared to 0.560 for **SciLitLLM1.5-14B**.

Table 6. Concept coverage analysis of model-generated summaries. Symbols: ✓= fully covered; ∼ = partially covered; × = not covered. Overall performance scores are derived using the weighted metric aggregation described in Section 2.4.3.

Reference Concept	mistral_medium-2505	SciLitLLM1.5-14B
H1: Switch-like response	✓	∼
H2: Surface density threshold	✓	✓
H3: Percolation phase transition	✓	×
H4: Discrete decision-making	✓	∼
Coverage Score	4.0 / 4.0	1.5 / 4.0
Semantic Similarity[a]	0.946	0.731
Overall Performance Score	0.619	0.560

[a] *Cosine similarity to highlights (all-mpnet-base-v2).*

4. Discussion

The benchmarking analysis revealed clear performance differences between the evaluated summarization approaches. Overall, general-purpose LLMs achieved the highest summarization quality across all surface-level and embedding-based metrics, followed closely by general-purpose SLMs and reasoning-oriented LLMs. In contrast, domain-specific models, encoder–decoder architectures such as T5 and PEGASUS, and traditional extractive methods like TextRank all reached noticeably lower performance levels. These results highlight the clear progression from extractive and encoder–decoder approaches toward transformer-based models, while also showing that domain-specific fine-tuning alone does not necessarily lead to improved summarization quality.

4.1. Model Group Comparisons

To understand the causes of the observed performance differences, the models were compared by architecture, size, and domain focus. This analysis examines how model scale, reasoning ability, and domain specialization influence summarization quality in biomedical texts. The next sections discuss these aspects in detail by comparing large and small language models, general-purpose and domain-specific models, and general-purpose and reasoning-oriented models, referring to Figure 1.

4.1.1. General-purpose Large vs. Small Language Models (LLMs vs. SLMs)

By comparing the overall performance of SLMs and LLMs, even among the top 10 of the best-performing models the majority are LLMs. This is likely due to their huge number of parameters which allow them to better understand the complex context typical for biomedical literature. While very small models, like Gemma3:270M, can indeed lack the capacity to handle this complexity, SLMs remain competitive, with some models even outperforming certain LLMs. This may be because smaller datasets are often more curated and of higher quality compared to the large amount of data required to train a big model [69].

Interestingly, medium-sized models in the range of approx. 20-70B parameters (e.g the Mistral family), appear to be more performant than larger proprietary ones. These models seem to reach an optimal compromise about number of parameters and overall performance where additional parameters could disrupt this equilibrium, leading to potentially over-fitting or plateauing performance [70].

4.1.2. General-purpose vs. Domain-specific Models

As discussed in the introduction section, domain-specific models have been developed over time with the aim of improving model ability for a specific task. However, in this article,

we found that overall general-purpose models outperform both domain-specific models specialized in the biomedical domain and the one specialized for text summarization, regardless of model size. A possible explanation for this behavior is that domain-specific models fine-tuned on biomedical text, might be better for learning and understanding the complex biomedical terminology or lexical patterns but might fail in summarization tasks. On the other hand, models specifically designed for text summarization might be good at summarizing in general but fail at capturing the complex biomedical meaning. That is why generalist models, leveraging their broad knowledge, seem to perform better [71]. Additionally, domain-specific models can "forget" the general knowledge that was acquired during the pre-training phase, experiencing a phenomenon called "catastrophic forgetting", which represents an issue when the task requires both domain-specific knowledge, the biomedical knowledge, and context understanding, for text summarization [72].

4.1.3. General-purpose vs Reasoning-oriented Models

Even though some reasoning-oriented models ranked among the top 10 of the best performing models, most of them were positioned in the middle of the ranking as moderate performing models. This can be explained by the intrinsic multi-step logical reasoning nature of these models, that while it can be advantageous for tasks that require breaking problems down into sequential steps like for mathematics or coding, it can be not ideal for text summarization that require semantic compression and factual grounding instead [73].

4.2. Evaluation and Metric Considerations

The evaluation framework combined surface-level, embedding-based, and factual consistency metrics to capture complementary aspects of summarization quality. Surface-level metrics such as ROUGE, BLEU, and METEOR primarily measure lexical overlap with the reference summaries, while embedding-based metrics including RoBERTa, DeBERTa, and MPNet assess semantic similarity and paraphrasing ability. AlignScore adds a distinct perspective by evaluating factual consistency between the generated summary and its source abstract. Unlike other metrics, it directly compares the summary with the input rather than with the human-written Highlights section. This design enables AlignScore to evaluate factual faithfulness to the source material instead of measuring the similarity to the reference summary.

While the correlation analysis indicated broad agreement among most metrics, AlignScore showed weaker alignments with the others, which emphasizes that factual consistency represents a distinct dimension of summarization quality. The strong performance of extractive approaches such as the frequency-based and TextRank models illustrates this difference: by retaining sentences from the abstract almost verbatim, these models naturally preserve factual accuracy and therefore achieve high AlignScore values, despite weaker results on other metrics.

Nevertheless, using AlignScore in this benchmark was intentional as factual grounding is an essential requirement in scientific text summarization. Models that generate fluent or semantically similar summaries may still introduce factual inaccuracies or exclude key information. Including AlginScore therefore ensures that the benchmark considers both linguistic quality and factual reliability.

4.3. Model Access Methods and API Heterogeneity

Model access methods varied across the evaluation due to differing API capabilities and requirements. HuggingFace models were accessed through their supported interfaces: the pipeline API (task="summarization") where available, or chat/completion formats for models that did not support the pipeline approach. Ollama models required use of the generate endpoint with merged prompts, while OpenAI, Anthropic, and Mistral models

each mandated their respective provider-specific APIs (responses.create, messages.create, and chat.complete) with distinct message structures. We applied hyperparameter normalization where possible, though API-level constraints prevented full standardization. For example, GPT-5 does not support temperature control, instead offering only reasoning-specific parameters. Additionally, proprietary middleware layers may transform requests and responses in undocumented ways, potentially affecting outputs independently of the underlying model architectures. These necessary methodological variations warrant consideration when interpreting performance differences across models.

4.4. Limitations and Future Work

This benchmark focuses on a single summarization task: generating concise summaries from biomedical abstracts. This setup provides a clear and well-defined evaluation framework, but the findings may not fully extend to other forms of scientific or biomedical summarization, including full-length articles, clinical trial data, or lay-oriented summaries. The rapid progress in LLMs means these results reflect a specific snapshot in time and may change as newer architectures and models become available.

Another limitation lies in the exclusive reliance on automatic evaluation metrics. Although combining surface-level, embedding-based, and factual measure offers a broad view, human assessment would provide a more nuanced understanding of readability, coherence, and factual correctness. Future work could therefore extend this benchmark by integrating expert-based evaluations, exploring alternative summarization tasks, and including emerging model families as they are released.

4.5. Practical Implications and Applications

The results of this benchmark provide useful guidance for selecting summarization models in biomedical and scientific settings. The strong performance of general-purpose LLMs indicates that broad, diverse pretraining is often more advantageous than narrow domain adaptation when dealing with unseen scientific content.

Another key consideration is the trade-off between output quality and processing efficiency. While LLMs achieved the highest overall scores, SLMs deliver competitive results at substantially lower computational cost, which makes them especially attractive for large-scale or resource-constrained applications. Choosing between large and small models therefore depends not only on desired output quality but also on the intended scale of summarization.

Overall, the findings suggest that general-purpose LLMs currently offer the most reliable and practical choice for biomedical summarization. Their consistent performance across evaluation criteria demonstrates that broad generalization outweighs the marginal gains from more narrowly specialized or fine-tuned approaches, many of which are not primarily optimized for summarization.

5. Conclusions

<TBD>

Draft: For scientific text summarization specifically, models need sufficient capacity to understand domain terminology and complex relationships without the computational overhead and potential overfitting of massive proprietary systems.

The semantic comprehension and text generation strengths of general-purpose LLMs outweigh the multi-step reasoning capabilities that reasoning-oriented models bring to other domains

References

1. Zhang, Y.; Jin, H.; Meng, D.; Wang, J.; Tan, J. A Comprehensive Survey on Process-Oriented Automatic Text Summarization with Exploration of LLM-Based Methods, 2025, [arXiv:cs.AI/2403.02901].
2. Zhang, H.; Yu, P.S.; Zhang, J. A Systematic Survey of Text Summarization: From Statistical Methods to Large Language Models, 2024, [arXiv:cs.CL/2406.11289].
3. Rohil, M.K.; Magotra, V. An exploratory study of automatic text summarization in biomedical and healthcare domain. *Healthcare Analytics* **2022**, *2*, 100058. https://doi.org/https://doi.org/10.1016/j.health.2022.100058.
4. Xie, Q.; Luo, Z.; Wang, B.; Ananiadou, S. A Survey for Biomedical Text Summarization: From Pre-trained to Large Language Models, 2023, [arXiv:cs.CL/2304.08763].
5. Luhn, H.P. The Automatic Creation of Literature Abstracts. *IBM J. Res. Dev.* **1958**, *2*, 159–165.
6. Edmundson, H.P. New Methods in Automatic Extracting. *J. ACM* **1969**, *16*, 264–285. https://doi.org/10.1145/321510.321519.
7. Robertson, S. Understanding Inverse Document Frequency: On Theoretical Arguments for IDF. *Journal of Documentation - J DOC* **2004**, *60*, 503–520. https://doi.org/10.1108/00220410410560582.
8. Reeve, L.H.; Han, H.; Nagori, S.V.; Yang, J.C.; Schwimmer, T.A.; Brooks, A.D. Concept frequency distribution in biomedical text summarization. In Proceedings of the Proceedings of the 15th ACM International Conference on Information and Knowledge Management, New York, NY, USA, 2006; CIKM '06, p. 604–611. https://doi.org/10.1145/1183614.1183701.
9. Mihalcea, R.; Tarau, P. TextRank: Bringing Order into Text. In Proceedings of the Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing; Lin, D.; Wu, D., Eds., Barcelona, Spain, 2004; pp. 404–411.
10. Shang, Y.; et al. Learning to rank-based gene summary extraction. *BMC Bioinformatics* **2014**, *15*, S10. https://doi.org/10.1186/1471-2105-15-S12-S10.
11. Erkan, G.; Radev, D.R. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research* **2004**, *22*, 457–479. https://doi.org/10.1613/jair.1523.
12. Afzal, M.; Alam, F.; Malik, K.M.; Malik, G.M. Clinical Context–Aware Biomedical Text Summarization Using Deep Neural Network: Model Development and Validation. *J Med Internet Res* **2020**, *22*, e19810. https://doi.org/10.2196/19810.
13. Almasoud, A.; Hassine, S.; Al-Wesabi, F.; Nour, M.; Hilal, A.; Al Duhayyim, M.; Hamza, A.; Motwakel, A. Automated Multi-Document Biomedical Text Summarization Using Deep Learning Model. *Computers, Materials & Continua* **2022**, *71*, 5800. https://doi.org/10.32604/cmc.2022.024556.
14. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need, 2023, [arXiv:cs.CL/1706.03762].
15. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019, [arXiv:cs.CL/1810.04805].
16. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *CoRR* **2019**, *abs/1910.13461*, [1910.13461].
17. Yuan, H.; Yuan, Z.; Gan, R.; Zhang, J.; Xie, Y.; Yu, S. BioBART: Pretraining and Evaluation of A Biomedical Generative Language Model. In Proceedings of the Proceedings of the 21st Workshop on Biomedical Language Processing; Demner-Fushman, D.; Cohen, K.B.; Ananiadou, S.; Tsujii, J., Eds., Dublin, Ireland, 2022; pp. 97–109. https://doi.org/10.18653/v1/2022.bionlp-1.9.
18. Abinaya, S.; Vigil, M.; Keerthika, K.; Varshasri, R. Medical Text Summarization Using BART with LoRA-Based Parameter Efficient Fine Tuning **2024**.
19. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, 2023, [arXiv:cs.LG/1910.10683].
20. Zhang, J.; Zhao, Y.; Saleh, M.; Liu, P.J. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization, 2020, [arXiv:cs.CL/1912.08777].
21. Beltagy, I.; Peters, M.E.; Cohan, A. Longformer: The Long-Document Transformer. *arXiv:2004.05150* **2020**.
22. Steblianko, O.; Shymkovych, V.; Kravets, P.; Novatskyi, A.; Shymkovych, L. Scientific article summarization model with unbounded input length. *Information, Computing and Intelligent systems* **2024**, pp. 150–158. https://doi.org/10.20535/2786-8729.5.2024.314724.
23. Plaat, A.; Wong, A.; Verberne, S.; Broekens, J.; van Stein, N.; Back, T. Multi-Step Reasoning with Large Language Models, a Survey, 2025, [arXiv:cs.AI/2407.11511].
24. Radford, A.; Narasimhan, K. Improving Language Understanding by Generative Pre-Training. 2018.
25. Bai, Y.; Kadavath, S.; Kundu, S.; Askell, A.; Kernion, J.; Jones, A.; Chen, A.; Goldie, A.; Mirhoseini, A.; McKinnon, C.; et al. Constitutional AI: Harmlessness from AI Feedback, 2022, [arXiv:cs.CL/2212.08073].
26. Team, G.; Kamath, A.; Ferret, J.; Pathak, S.; Vieillard, N.; Merhej, R.; Perrin, S.; Matejovicova, T.; Ramé, A.; Rivière, M.; et al. Gemma 3 Technical Report, 2025, [arXiv:cs.CL/2503.19786].

27. Turbitt, O.; Bevan, R.; Aboshokor, M. MDC at BioLaySumm Task 1: Evaluating GPT Models for Biomedical Lay Summarization. In Proceedings of the Proceedings of the 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks; Demner-fushman, D.; Ananiadou, S.; Cohen, K., Eds., Toronto, Canada, 2023; pp. 611–619. https://doi.org/10.18653/v1/2023.bionlp-1.65.
28. Li, S.; Huang, J.; Zhuang, J.; Shi, Y.; Cai, X.; Xu, M.; Wang, X.; Zhang, L.; Ke, G.; Cai, H. SciLitLLM: How to Adapt LLMs for Scientific Literature Understanding, 2025, [arXiv:cs.LG/2408.15545].
29. Wang, C.; Kantarcioglu, M. A Review of DeepSeek Models' Key Innovative Techniques, 2025, [arXiv:cs.LG/2503.11486].
30. Elsevier. Highlights, 2024. Accessed: 2025-08-07.
31. Cell Press. Final Submission: Other Components: Highlights, 2024. Accessed: 2025-08-07.
32. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* **2020**, *21*, 1–67.
33. Hasan, T.; Bhattacharjee, A.; Islam, M.S.; Mubasshir, K.; Li, Y.F.; Kang, Y.B.; Rahman, M.S.; Shahriyar, R. XL-Sum: Large-Scale Multilingual Abstractive Summarization for 44 Languages. In Proceedings of the Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Online, 2021; pp. 4693–4703.
34. Belcak, P.; Heinrich, G.; Diao, S.; Fu, Y.; Dong, X.; Muralidharan, S.; Lin, Y.C.; Molchanov, P. Small Language Models are the Future of Agentic AI, 2025, [arXiv:cs.AI/2506.02153].
35. Mishra, M.; Stallone, M.; Zhang, G.; Shen, Y.; Prasad, A.; Soria, A.M.; Merler, M.; Selvam, P.; Surendran, S.; Singh, S.; et al. Granite Code Models: A Family of Open Foundation Models for Code Intelligence, 2024, [arXiv:cs.AI/2405.04324].
36. Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. The Llama 3 Herd of Models, 2024, [arXiv:cs.AI/2407.21783].
37. Jiang, A.Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D.S.; de las Casas, D.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. Mistral 7B, 2023, [arXiv:cs.CL/2310.06825].
38. Abdin, M.; Aneja, J.; Awadalla, H.; Awadallah, A.; Awan, A.A.; Bach, N.; Bahree, A.; Bakhtiari, A.; Bao, J.; Behl, H.; et al. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone, 2024, [arXiv:cs.CL/2404.14219].
39. Abdin, M.; Aneja, J.; Behl, H.; Bubeck, S.; Eldan, R.; Gunasekar, S.; Harrison, M.; Hewett, R.J.; Javaheripi, M.; Kauffmann, P.; et al. Phi-4 Technical Report, 2024, [arXiv:cs.CL/2412.08905].
40. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners, 2020, [arXiv:cs.CL/2005.14165].
41. OpenAI.; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; et al. GPT-4 Technical Report, 2024, [arXiv:cs.CL/2303.08774].
42. Anthropic. Claude - Models overview, 2025. Accessed: 2025-09-24.
43. Hernández-Cano, A.; Hägele, A.; Huang, A.H.; Romanou, A.; Solergibert, A.J.; Pasztor, B.; Messmer, B.; Garbaya, D.; Ďurech, E.F.; Hakimi, I.; et al. Apertus: Democratizing Open and Compliant LLMs for Global Language Environments. https://arxiv.org/abs/2509.14233, 2025.
44. DeepSeek-AI.; Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; et al. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning, 2025, [arXiv:cs.CL/2501.12948].
45. Team, Q. Qwen3 Technical Report, 2025, [arXiv:cs.CL/2505.09388].
46. OpenAI. GPT-OSS: Open Source GPT Models, 2025. Accessed: 2025-09-23.
47. OpenAI. GPT-5 Models, 2025. Accessed: 2025-09-24.
48. Mistral-AI.; :.; Rastogi, A.; Jiang, A.Q.; Lo, A.; Berrada, G.; Lample, G.; Rute, J.; Barmentlo, J.; Yadav, K.; et al. Magistral, 2025, [arXiv:cs.CL/2506.10910].
49. Luo, R.; Sun, L.; Xia, Y.; Qin, T.; Zhang, S.; Poon, H.; Liu, T.Y. BioGPT: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics* **2022**, *23*, [https://academic.oup.com/bib/article-pdf/23/6/bbac409/47144271/bbac409.pdf]. bbac409, https://doi.org/10.1093/bib/bbac409.
50. Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open Foundation and Fine-Tuned Chat Models, 2023, [arXiv:cs.CL/2307.09288].
51. Ankit Pal, M.S. OpenBioLLMs: Advancing Open-Source Large Language Models for Healthcare and Life Sciences. https://huggingface.co/aaditya/OpenBioLLM-Llama3-70B, 2024.
52. Labrak, Y.; Bazoge, A.; Morin, E.; Gourraud, P.A.; Rouvier, M.; Dufour, R. BioMistral: A Collection of Open-Source Pretrained Large Language Models for Medical Domains, 2024, [arXiv:cs.CL/2402.10373].
53. Li, S.; Huang, J.; Zhuang, J.; Shi, Y.; Cai, X.; Xu, M.; Wang, X.; Zhang, L.; Ke, G.; Cai, H. SciLitLLM: How to Adapt LLMs for Scientific Literature Understanding, 2024, [arXiv:cs.LG/2408.15545].
54. Lin, C.Y. ROUGE: A Package for Automatic Evaluation of Summaries. In Proceedings of the Text Summarization Branches Out, Barcelona, Spain, 2004; pp. 74–81.

55. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: a Method for Automatic Evaluation of Machine Translation. In Proceedings of the Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics; Isabelle, P.; Charniak, E.; Lin, D., Eds., Philadelphia, Pennsylvania, USA, 2002; pp. 311–318. https://doi.org/10.3115/1073083.1073135.
56. Banerjee, S.; Lavie, A. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In Proceedings of the Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization; Goldstein, J.; Lavie, A.; Lin, C.Y.; Voss, C., Eds., Ann Arbor, Michigan, 2005; pp. 65–72.
57. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach, 2019, [arXiv:cs.CL/1907.11692].
58. He, P.; Liu, X.; Gao, J.; Chen, W. DeBERTa: Decoding-enhanced BERT with Disentangled Attention, 2021, [arXiv:cs.CL/2006.03654].
59. Song, K.; Tan, X.; Qin, T.; Lu, J.; Liu, T.Y. MPNet: Masked and Permuted Pre-training for Language Understanding, 2020, [arXiv:cs.CL/2004.09297].
60. Zha, Y.; Yang, Y.; Li, R.; Hu, Z. AlignScore: Evaluating Factual Consistency with a Unified Alignment Function, 2023, [arXiv:cs.CL/2305.16739].
61. McKinney, W. Data Structures for Statistical Computing in Python. In Proceedings of the Proceedings of the 9th Python in Science Conference; van der Walt, S.; Millman, J., Eds., 2010, pp. 51 – 56.
62. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.
63. Hagberg, A.; Swart, P.; S Chult, D. Exploring network structure, dynamics, and function using NetworkX. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.
64. Brin, S.; Page, L. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems* **1998**, *30*, 107–117. Proceedings of the Seventh International World Wide Web Conference, https://doi.org/https://doi.org/10.1016/S0169-7552(98)00110-X.
65. Bird, S.; Klein, E.; Loper, E. *Natural language processing with Python: analyzing text with the natural language toolkit*; " O'Reilly Media, Inc.", 2009.
66. Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K.Q.; Artzi, Y. BERTScore: Evaluating Text Generation with BERT, 2020, [arXiv:cs.CL/1904.09675].
67. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, 2019, [arXiv:cs.CL/1908.10084].
68. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Online, 2020; pp. 38–45.
69. Gunasekar, S.; Zhang, Y.; Aneja, J.; Mendes, C.C.T.; Giorno, A.D.; Gopi, S.; Javaheripi, M.; Kauffmann, P.; de Rosa, G.; Saarikivi, O.; et al. Textbooks Are All You Need, 2023, [arXiv:cs.CL/2306.11644].
70. Zhou, L.; Schellaert, W.; Plumed, F.; Moros-Daval, Y.; Ferri, C.; Hernández-Orallo, J. Larger and more instructable language models become less reliable. *Nature* **2024**, *634*, 61–68. https://doi.org/10.1038/s41586-024-07930-y.
71. Dorfner, F.J.; Dada, A.; Busch, F.; Makowski, M.R.; Han, T.; Truhn, D.; Kleesiek, J.; Sushil, M.; Lammert, J.; Adams, L.C.; et al. Biomedical Large Languages Models Seem not to be Superior to Generalist Models on Unseen Medical Data, 2024, [arXiv:cs.CL/2408.13833].
72. Zhai, Y.; Tong, S.; Li, X.; Cai, M.; Qu, Q.; Lee, Y.J.; Ma, Y. Investigating the Catastrophic Forgetting in Multimodal Large Language Models, 2023, [arXiv:cs.CL/2309.10313].
73. Jin, K.; Wang, Y.; Santos, L.; Fang, T.; Yang, X.; Im, S.K.; Oliveira, H.G. Reasoning or Not? A Comprehensive Evaluation of Reasoning LLMs for Dialogue Summarization, 2025, [arXiv:cs.CL/2507.02145].